DTU Library

A PID autotuner utilizing GPC and constraint optimization

Henningsen, Arne; Christensen, Anders; Ravn, Ole

Published in:
Proceedings of the 29th IEEE Conference on Decision and Control

Link to article, DOI:
10.1109/CDC.1990.203857

Publication date:
1990

Document Version
Publisher's PDF, also known as Version of record

Proceedings of the 29th Conference
on Decision and Control
Honolulu, Hawaii • December 1990

A PID Autotuner Utilizing GPC and Constraint Optimization.

Arne Henningsen, M.Sc., Ass. Prof,
Anders Christensen, M.Sc., Assoc. Prof,
Ole Ravn, M.Sc., Assoc. Prof.

Institute of Automatic Control Systems, Technical University of Denmark, Bldg. 326
DK-2800 Lyngby, Denmark

Abstract: This paper offers a solution to the PID autotuning problem, by constraining the parameters of a discrete 2nd order discrete-time controller. The integrator is forced into the regulator by using a CARIMA model. The discrete-time regulator parameters are calculated by optimizing a GPC criterion, and the PID structure is ensured by constraining the parameters to a feasible set defined by the discrete-time Euler approximation of the ideal continuous-time PID controller. The algorithm is extended by incorporating constraints in amplitude and slew-rate of the control signal. Generality is discussed, and some of the significant properties of the algorithm is shown by simulations.

Keywords: GPC, adaptive control, PID control.

Introduction

One of the classic aims for adaptive control is to solve the problem of designing a general-purpose regulator, that can be applied to a large class of industrial processes without regard to the modelling of the process.

The PID controller has, due to its simplicity and robustness, been used extensively in industry to control a large number of processes. It is commonly recognized, that industrial controllers of the PID type are often operating with poor tuning, partly due to the large time constants in many processes and partly to the lack of on-location expertise. Therefore, the problem of automatic tuning of the PID structure has been straightforward and offered much investigation.

A well-known successful solution to the problem has been developed by Hägglund and Åström [1] and implemented in the SattControl PID autotuner, which is today commercially available in 3rd version. This regulator is tuned on an experimental basis, by inserting a relay into the control loop. This relay is activated by the tuning button and provokes the system to oscillate in the negative real axis intersection point of the Nyquist curve. Using the Ziegler-Nichols rule, the continuous-time controller parameters are determined. One of the niceties about this approach is that it is able to tune differential action on time constants of several hours.

Cameron and Seborg [2] introduces the idea of translating a discrete-time direct 2nd order adaptive controller to a set of continuous-time parameters. This approach is designated an "adaptive controller with a PID structure". The generalized minimum variance technique is used to design a direct adaptive 2nd order controller. Integral action is obtained by using the fact that the denominator polynomium of the estimated regulator can be "set" to an integrator by manipulating the control signal weighting. The numerator polynomial is then translated by an Euler approximation of the 2nd order continuous-time operator to give a set of equations uniquely determining an equivalent PID regulator.

Vega and Zarrop [3] introduces an LQG type performance index to optimize the discrete-time regulator parameters. This is an indirect approach where the performance index is calculated from the estimated model of the system, and afterwards the controller parameters are iteratively calculated to minimize the index.

Inspired by these approaches, this paper offers a PID autotuning principle which is based on an indirect adaptive GPC controller.

Features of the proposed method are:

- CARIMA model
- Any parameter estimator
- Recursive solution of the Diophantine equations for calculating a GPC criterion
- Design by minimization of the GPC criterion with respect to a PID structure discrete-time controller, using constraint optimization with respect to controller parameters and control signal limits.

The choice of a GPC-type criterion is natural due to the celebrated robustness properties, and thus fits well to the robust performance of the PID type controllers. The GPC approach by Clarke et al. [4] is based on a CARIMA model, which forces an integrator into the controller. The system parameters are identified to calculate the criterion entries, and the Cameron & Seborg PID structure controller [2] is used for the minimization. The optimization of the criterion to determine the discrete-time regulator parameters is done recursively by quadratic programming, obtaining an RLS-like algorithm to the unconstrained solution. In order to obtain a set of discrete-time parameters that resembles a continuous-time PID controller, constraints on the discrete-time controller parameters are calculated and incorporated into the optimization algorithm. We furthermore include slew-rate and amplitude constraints on the control signal.

The cautiousness of the solution is argued and shown to depend on the frequency contents of the reference signal and on the control signal limits. The total algorithm is stated with design parameters and the generality is discussed. Operational characteristics of this algorithm are shown by simulation studies.

Process model and output prediction

Assume that the process dynamics can be represented by the SISO discrete-time CARIMA model

$$A(q^{-1})y(t) = B(q^{-1})u(t-1) + C(q^{-1})\varepsilon(t)/(1-q^{-1}) \tag{1}$$

where A,B and C are polynomials in the backward shift operator q^{-1},

$$A(q^{-1}) = 1 + a_1q^{-1} + ... + a_{na}q^{-na}$$
$$B(q^{-1}) = b_0 + b_1q^{-1} + ... + b_{nb}q^{-nb}$$
$$C(q^{-1}) = 1 + c_1q^{-1} + ... + c_{nc}q^{-nc}$$

If the process has a non-zero deadtime the leading elements of the polynomial $B(q^{-1})$ are zero. In (1), y(t) is the measured output, u(t) is the control input and $\varepsilon(t)$ is a zero mean white noise sequence. For simplicity in the following development $C(q^{-1})$ is chosen to be 1.

Based on the CARIMA model, prediction of future outputs is obtained as described by Clarke et al. [4]. To derive a j-step predictor of y(t+j) one uses the identity

$$1 = E_j(q^{-1})A(1-q^{-1}) + q^{-j}F_j(q^{-1}) \tag{2}$$

where E_j and F_j are polynomials uniquely defined given $A(q^{-1})$ and the prediction horizon j. If (1) is multiplied by $E_j(1-q^{-1})q^j$ and $E_jA(1-q^{-1})$ is substituted from (2) we have

CH2917-3/90/0000-1475$1.00 © 1990 IEEE

$$y(t+j) = E_jBdu(t+j-1) + F_jy(t) + E_j\varepsilon(t+j) \quad (3)$$

and the optimal predictor is given by

$$\hat{y}(t+j \mid t) = G_jdu(t+j-1) + F_jy(t) \quad (4)$$

where $G_j(q^{-1}) = E_jB = G_{j0} + G_{j1}q^{-1} + G_{j2}q^{-2} + \ldots\ldots + G_{jj}q^{-j} + ..$

Clarke et al. [4] suggests solving the Diophantine equations recursively, which is much simpler than using a separate predictor for each output horizon.

PID control

An ideal PID controller is given by

$$u(t) = K_c \left[e(t) + \frac{1}{T_i}\int_0^t e(\tau)\, d\tau + T_d \frac{de(t)}{dt} \right] \quad (5)$$

where K_c, T_i and T_d are the gain, the integral time and the derivative time respectively. At time t, u(t) is the control signal, and the error signal e(t) is given as

$$e(t) \equiv w(t) - y(t),$$

where w(t) is the reference signal, and y(t) is the system output. Discretizing (5) with sampling period T_s gives, using a first order approximation, the following ideal digital PID controller written in velocity form (Cameron & Seborg [2])

$$du(t) = K_c\{e(t) - e(t-1) + \frac{T_s}{T_i} e(t) + \frac{T_d}{T_s}[e(t)-2e(t-1)+e(t-2)]\} \quad (6)$$

where $du(t) \equiv u(t) - u(t-1)$ and K_c, T_i and T_d are identical to the continuous-time PID controller settings.
By defining a vector of control errors $\underline{e}_t$

$$\underline{e}_t = [e(t)\ e(t-1)\ e(t-2)]^T$$

and a vector of controller parameters $\underline{g}$

$$\underline{g} = [\ g_0\ g_1\ g_2\ ]^T$$

a general three-term controller can be written in the form

$$du(t) = g_0e(t) + g_1e(t-1) + g_2e(t-2) = \underline{g}^T\underline{e}_t \quad (7)$$

The controller (7) is seen to describe an ideal PID controller with the controller settings

$$K_c = -(\ g_1 + 2g_2\ )$$

$$T_d = \frac{-g_2T_s}{g_1 + 2g_2}$$

$$T_i = -T_s \frac{g_1 + 2g_2}{g_0 + g_1 + g_2}$$

The PID controller settings must be finite, positive scalars

$$K_c > 0 \qquad T_i > 0 \qquad T_d > 0.$$

This constrains the feasible controller parameters to the region Ω, given by

$$\begin{aligned} g_1 + 2g_2 &\le -\varepsilon_1 < 0 \\ g_2 &\ge 0 \\ g_0 + g_1 + g_2 &\ge \varepsilon_2 > 0, \end{aligned} \quad (8)$$

where ε_1 and ε_2 are small positive scalars.

A drawback of representing the PID controller in the velocity form is that it cannot operate in P- or PD-mode because of the inherited integral action. If the integral action is not needed the integrator must be compensated. This means that an unstable mode has to be canceled and can lead to difficulties.
An advantage of representing the PID controller in the velocity form is that the controller will not suffer from the problems due to reset-windup, because the integration will stop automatically if the output is limited.
Remark 1: The allowed regions for g_0, g_1 and g_2 are derived in the case that $K_c > 0$. If $K_c < 0$, the regions must be modified accordingly.

Signal limitations and constrained solutions

In practical applications control signals will always be limited due to the physical constraints imposed by the actuator.
The constraints can be of slew-rate or amplitude type. The slew-rate constraint reflects the ability of the actuator to handle changes in the control signal. Let $du_0(t)$ be the computed control signal and du_{min} and du_{max} the slew-rate constraints imposed by the actuator. Then the slew-rate constrained signal du(t) is given by

$$\begin{aligned} du(t) &= du_{min} &&: \quad du_0(t) \le du_{min} \\ du(t) &= du_0(t) &&: \quad du_{min} \le du_0(t) \le du_{max} \\ du(t) &= du_{max} &&: \quad du_{max} \le du_0(t) \end{aligned}$$

The amplitude constraint is caused by the maximum capacity of the actuator. Given constraints u_{min} and u_{max}, the amplitude constrained control signal u(t) is

$$\begin{aligned} u(t) &= u_{min} &&: \quad u_0(t) \le u_{min} \\ u(t) &= u_0(t) &&: \quad u_{min} \le u_0(t) \le u_{max} \\ u(t) &= u_{max} &&: \quad u_{max} \le u_0(t) \end{aligned}$$

The quadratic programming problem used in this paper incorporates the control signal constraints within (8), the feasible region of the controller parameters. From (7), we have

$$du(t) = g_0e(t) + g_1e(t-1) + g_2e(t-2).$$

The slew-rate constraint, specified as

$$du_{min} \le du(t) \le du_{max},$$

is directly related to the controller parameters. Regarding the amplitude constraint, use

$$u(t) = u(t-1) + du(t) \text{ and } u_{min} \le u(t) \le u_{max}$$

to obtain

$$u_{min} - u(t-1) \le du(t) \le u_{max} - u(t-1).$$

Given the control error and u(t-1), the signal constraints can now be reformulated in terms of the PID controller parameters by the feasible region Σ, given by

$$\begin{aligned} du_{min} \le g_0e(t) + g_1e(t-1) + g_2e(t-2) &\le du_{max} \\ u_{min} - u(t-1) \le g_0e(t) + g_1e(t-1) + g_2e(t-2) &\le u_{max} - u(t-1) \end{aligned}$$

Define the region Ω_t of feasible controller parameters $\underline{g}$ at time t. This region satifies both the PID structure constraints and the control signal constraints, iff

$$\Omega_t = (\Omega \cup \Sigma)$$

or

$$\begin{gathered} g_1 + 2g_2 \leq -\varepsilon_1 \\ g_2 \geq 0 \\ g_0 + g_0 + g_2 \geq \varepsilon_2 \\ du_{min} \leq g_0 e(t) + g_1 e(t-1) + g_2 e(t-2) \leq du_{max} \\ u_{min} - u(t-1) \leq g_0 e(t) + g_1 e(t-1) + g_2 e(t-2) \leq u_{max} - u(t-1) \end{gathered} \tag{9}$$

Adaptive PID control

In deriving a self-tuning PID controller a method greatly inspired by the method proposed by Vega & Zarrop [3] will be considered. Using the GPC criterion and taking into account signal limitations the proposed design is described in figure 1.

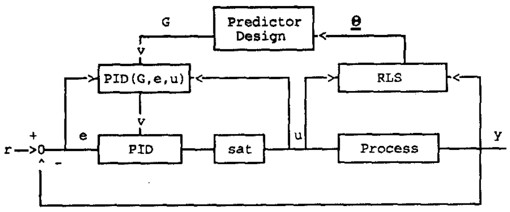

Figure 1. Diagram showing the principles of the self-tuning PID controller. The method is based on a certainty equivalence indirect approach. Taking signal limitations into account, the PID parameters are determined recursively given the output predictors G, the control error e and the previous control signal u(t-1).

Thus using a certainty equivalence indirect approach the following scheme is proposed:

At each sampling instant the following three steps are performed

1. Estimate the system parameters A and B getting $\underline{\Theta}$
2. Determine the region Ω_t of feasible solutions from (9).
3. Determine using quadratic programming the controller parameters $\underline{g}$ as

$$\underline{g} = \arg \min_{\underline{g} \in \Omega t} J(\underline{g}, \underline{\Theta})$$

where the cost function J is determined by the GPC criterion.

Consider the GPC criterion

$$J = E\ \{\ \sum_{j=1}^{N} [y(t+j) - w(t+j)]^2 + \alpha \sum_{j=1}^{N_u} du(t+j-1)^2\} \tag{10}$$

where N is the maximum costing horizon;
N_u is the control horizon, and
α is a control-weighting constant.

w(t+j) denotes the desired reference signal at time t+j. If the future reference signals are not known it can be assumed that w(t+j) = w(t).

The problem is to minimize the GPC criterion under the restriction that the resulting controller must be a PID controller. As the PID controller determines du(t), but no future control signals, the control horizon N_u must be chosen to 1. This gives the following reduced GPC criterion to be minimized with respect to the controller parameters $\underline{g}$

$$J(\underline{g}) = E\ \{\ \sum_{j=1}^{N} [y(t+j) - w(t+j)]^2 + \alpha\, du(t)^2\ \} \tag{11}$$

$$J(\underline{g}) = \sum_{j=1}^{N} [\hat{y}(t+j\,|\,t) - w(t+j)]^2 + \alpha\, du(t)^2 \tag{12}$$

Using the predictors derived as in (2) - (4) but calculated based on the parameter estimates $\underline{\Theta}$ gives

$$J(\underline{g},\underline{\Theta}) = \sum_{j=1}^{N} [\ G_j du(t+j-1) + F_j y(t) - w(t+j)\ ]^2 + \alpha\, du(t)^2 \tag{13}$$

where du(t+j-1) = 0 j>1.

At time t the problem of determining the controller parameters $\underline{g}(t)$ can be formulated as a quadratic programming problem

$$\underline{g} = \arg \min J(\underline{g},\underline{\Theta}) = \arg \min [\underline{b}_t^T \underline{g} + \tfrac{1}{2} \underline{g}^T C_t \underline{g}] \tag{14}$$

Applying unconstrained optimization the criterion is minimized by solving

$$\underline{b}_t + C_t \underline{g} = 0 \tag{15}$$

giving the controller parameters $\underline{g}$.

Inserting the GPC criterion $\underline{b}_t$ and C_t are given as

$$\underline{b}_t = \pi_b(t)\ \underline{e}_t, \text{ and } C_t = \pi_c(t)\ \underline{e}_t \underline{e}_t^T$$

where $\pi_b(t)$ and $\pi_c(t)$ can be computed as

$$\pi_b(t) = 2 \sum_{j=1}^{N} G_{j,j-1}\ [\ \hat{y}(t+j\,|\,t) - G_{j,j-1} du(t) - w(t+j)\ ] \tag{16}$$

$$\pi_c(t) = 2\ [\ \sum_{j=1}^{N} (G_{j,j-1})^2 + \alpha\ ] \tag{17}$$

It is seen that $\pi_b(t)$ reflects the control error while $\pi_c(t)$ reflects the weighting of the control signal.

Computing again the unconstrained solution we get

$$\underline{g} = -\ \frac{\pi_b(t)}{\pi_c(t)}\ (\underline{e}_t \underline{e}_t^T)^{-1} \underline{e}_t$$

Having determined the unconstrained solution $\underline{g}$ a quadratic programming algorithm is used to ensure that a optimal solution $\underline{g}_C$ which lies within Ω_t is obtained. Given the constrained solution $\underline{g}_C$, the unconstrained solution $\underline{g}$ is replaced by $\underline{g}_C$ in the recursive update of $\underline{g}$.

PID autotuning

As the method is to be used as PID tuning experiment the criterion has to be minimized with respect to all previous information. This implies, that the controller parameters $\underline{g}$ has to be determined such that

$$\sum_{t=1}^{T} \underline{b}_t + \sum_{t=1}^{T} C_t \underline{g}(t) = 0 \tag{18}$$

is solved at every sampling instant.

Assuming that no constraints are imposed the solution to the autotuning problem can be determined as

$$g(t) = -\begin{bmatrix} \sum_{t=1}^{T} C_t \end{bmatrix}^{-1} * \begin{bmatrix} \sum_{t=1}^{T} \underline{b}_t \end{bmatrix} \tag{19}$$

Defining the gain matrix $\mathbf{P}(t)$ as

$$\mathbf{P}(t) = -\begin{bmatrix} \sum_{t=1}^{T} C_t \end{bmatrix}^{-1} \tag{20}$$

the solution to (18) can be written as

$$g(t) = \mathbf{P}(t) * \sum_{t=1}^{T} \underline{b}_t \tag{21}$$

According to the normal procedures when deriving the RLS algorithm using the matrix inversion lemma (ex. Goodwin & Sin, [5]), the solution to (21) can be computed recursively as

$$g(t) = g(t-1) - \frac{\pi_c(t)\mathbf{P}(t-1)\underline{e}_t}{1 + \pi_c(t)\underline{e}_t^T\mathbf{P}(t-1)\underline{e}_t} \left[\underline{e}_t^T g(t-1) + \frac{\pi_b(t)}{\pi_c(t)}\right] \tag{22}$$

$$\mathbf{P}(t) = \mathbf{P}(t-1) - \frac{\pi_c(t)\mathbf{P}(t-1)\underline{e}_t\underline{e}_t^T\mathbf{P}(t-1)}{1 + \pi_c(t)\underline{e}_t^T\mathbf{P}(t-1)\underline{e}_t} \tag{23}$$

Thus in the case of no constraints it is possible to derive the controller parameters in a way very similar to the RLS update. In this case the update is driven by the difference between the predicted control signal given by $\underline{e}_t^T g(t-1)$ and the optimal control signal reflected by $\pi_b(t)$ and $\pi_c(t)$. Note that the gain matrix $\mathbf{P}(t)$ will go to zero as the accumulated control error goes to infinity.

The algorithm

To summarize we give the proposed autotuning algorithm.
At each sampling instant

1. Using RLS, estimate the system parameters A and B getting $\underline{\Theta}$.

2. Using $\underline{\Theta}$, compute the output predictors from the recursive diophantine equation

$$1 = E_j(q^{-1})A(1-q^{-1}) + q^{-j}F_j(q^{-1}) \tag{2}$$

$$y\hat{y}(t+j\,|\,t) = G_j du(t+j-1) + F_j y(t) \tag{4}$$

3. As given by the GPC criterion compute $\pi_b(t)$ and $\pi_c(t)$ given by

$$\pi_b(t) = 2\sum_{j=1}^{N} G_{j,j-1}\,[\,\hat{y}(t+j\,|\,t) - G_{j,j-1}du(t) - w(t+j)\,] \tag{16}$$

$$\pi_c(t) = 2\,[\,\sum_{j=1}^{N} (G_{j,j-1})^2 + \alpha\,] \tag{17}$$

4. Determine the feasible region Ω_t from

$$\begin{gathered} g_1 + 2g_2 \le -\varepsilon_1 \\ g_2 \ge 0 \\ g_0 + g_1 + g_2 \ge \varepsilon_2 \\ du_{min} \le g_0 e(t) + g_1 e(t-1) + g_2 e(t-2) \le du_{max} \\ u_{min}-u(t-1) \le g_0 e(t) + g_1 e(t-1) + g_2 e(t-2) \le u_{max}-u(t-1) \end{gathered} \tag{9}$$

5. Determine the unconstrained solution $\underline{g}$ from

$$g(t) = g(t-1) - \frac{\mathbf{P}(t-1)\underline{e}_t}{1 + \pi_c(t)\underline{e}_t^T\mathbf{P}(t-1)\underline{e}_t} [\pi_c(t)\underline{e}_t^T g(t-1) + \pi_b(t)] \tag{22}$$

$$\mathbf{P}(t) = \mathbf{P}(t-1) - \frac{\pi_c(t)\mathbf{P}(t-1)\underline{e}_t\underline{e}_t^T\mathbf{P}(t-1)}{1 + \pi_c(t)\underline{e}_t^T\mathbf{P}(t-1)\underline{e}_t}, \tag{23}$$

then use the QP programming algorithm to determine $\underline{g}_C$.

6. Compute the control signal u(t) as $u(t) = u(t-1) + \underline{g}^T\underline{e}_t$

Discussion

Stability

Stability of the closed loop system has not been proved, however some heuristic arguments will be given. As the RLS estimator is stable, convergence of the estimator is assured. Assuming persistantly exiting signals and correct model order, the parameters will converge to the values of the true system. Now, if the process can be stabilized by a PID controller, that is, if the error signal is bounded, the controller parameters will converge as the **P** matrix goes to zero. The closed loop system will be stable. The GPC criterion (and LRP) yields robustness against overparameterization and impacts of model uncertainties are reduced. Thus stability of the closed loop system is not crucially depending on the convergence of the model parameters to the values of the true system, and the somewhat unrealistic demand of correct model order is of relaxed importance.

Signal limitations

In autotuning, a time-invariant PID controller is tuned. The gain of this resulting controller will be such that the control signals never saturate. This is undesirable from a time optimality point of view. The advantages of taking signal limitations into account are primarily for time-varying controllers, where the controller gain can be continuously adapted to the signal constraints.

The control signal can be prevented from saturation by reducing the high-frequency content of the reference signal. This reduction is achieved by specifying the desired bandwidth by a reference model and incorporating this as predictor filters.

Alternatively, the signal constraints in the control algorithm can be chosen such that saturation of the control signal is allowed for a small period. Then, the resulting controller compromises between time-optimality and speed of convergence.

Quadratic programming considerations

The chosen QP algorithm used must be started with a feasible set of controller parameters (inside Ω_t). As Ω_t changes every sampling instant, special precautions have to be taken when determining the starting point. If there is no feasible starting point the parameter update is suspended. Alternatively, algorithms with built-in starting points can be used, see e.g. [7,8].

Processes with dead-time

Using Long-Range Prediction it is possible to tune a PID controller for a system with unknown dead-time, provided that a closed loop stable solution exists.

This controller is generally of a poor quality, because the control tends to be very sluggish. If this is the case, dead-time compensation using a Smith predictor can improve performance. In this case the tuning should produce a PID controller, an estimate of the process model and an estimate of the dead-time of the process.

User choices

The significant design parameters for tuning the algorithm are

- sampling period T_s
- model order
- prediction horizon N
- control signal weighting α
- amplitude control signal constraint u_{max}
- slew-rate control signal constraint du_{max}
- predictor bandwidth adjustment

When considering the performance of a digital controller, the choice of sampling period is crucial. In our approach, the sampling period must be determined by the user, thus making the controller less general, requiring some tuning expertise.

The prediction horizon N determines a tradeoff between robustness and performance of the design. The closed-loop system is robust with high N (long range prediction), and bandwidth increases with decreasing N. If the controller is to be generally applicable the long-range prediction should be used. Given N, the control signal weighting α penalizes the bandwidth of the control loop. In practical applications, prediction horizon will be determined by robustness considerations and bandwidth afterwards tuned with α. Thus, the tuning of N requires control expertise, while control weighting should be available to the operator, enabling him to obtain a desired system performance.

The control signal constraints u_{max} and du_{max} should be chosen to reflect the physical limitations of the system.

If signal limitations are causing the tuned controller to be undesirably slow the predictor bandwidth can be adjusted. This reduces the highfrequency content of the reference signal in the controller design optimization, giving a faster closed loop system. Thus a prediction bandwidth button could be very helpful to the operator. Note that this feature must be inserted in the presented algorithm. Insertion of predictor bandwidth is a standard method, see e.g. Clarke et al. [4].

Simulation results

Properties of the proposed autotuning algorithm are illustrated using the standard level control of two coupled tanks. For a detailed description of the process, see e.g. [6].

Choosing a steady state flow the system equations can be linearized around the steady state values of the liquid levels h_1 and h_2. This leads to the transfer function from the controlled input flow, q_v (to tank 1) to the level h_2 in tank 2

$$\frac{h_2(s)}{u(s)} = K_P \frac{k1}{s^2 + s(k_1-k_2+k_3) + k_1k_3} \tag{24}$$

with $k_1 = 0.0625$, $k_2 = -0.0625$, $k_3 = 0.125$ and $K_P = 0.1$. Discretizing with sampling time T = 1 sec., we get

$$H_T(z^{-1}) = \frac{0.005313z^{-1} + 0.004498z^{-2}}{1 - 1.582z^{-1} + 0.6065z^{-2}} \tag{25}$$

The prediction horizon and the control signal weighting are chosen as N = 15 and α = 0.1. The following figures illustrate the performance of the autotuner with no control saturation, with control saturation, including the saturation as constraints in the autotuner, and limiting the bandwidth of the predictors ni the autotuner by a reference model. The resulting parameters are listed in table 1.

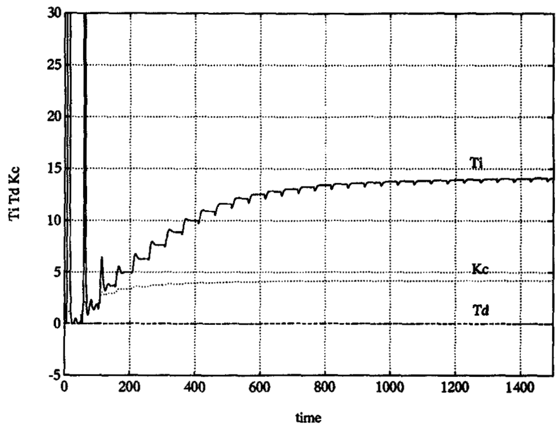

Figure 2: Autotuning without constraints, PID parameters.

Although minor changes are found in T_i, figure 2 shows convergence of the controller parameters in the linear case with no saturation. A nice property of the algorithm is that T_d converges to zero, reflecting that derivative action is not needed in this case. At this point, note that as integration is included in the model description, Ti can not go to zero. Difficulties will arise when tuning for a type 1 system. (in this case Ti must be limited, as the algorithm will try to compensate the integrator).

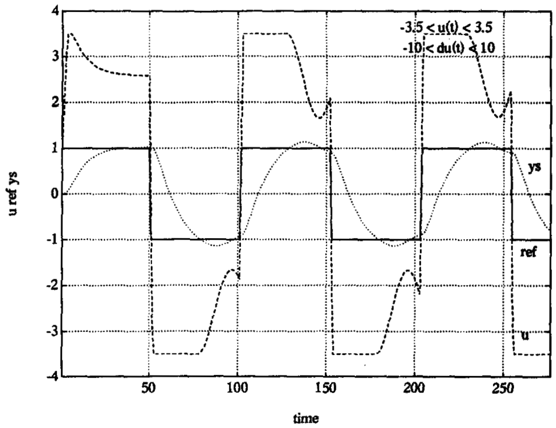

Figure 3: Autotuning with control signal saturation: reference, input and output signals.

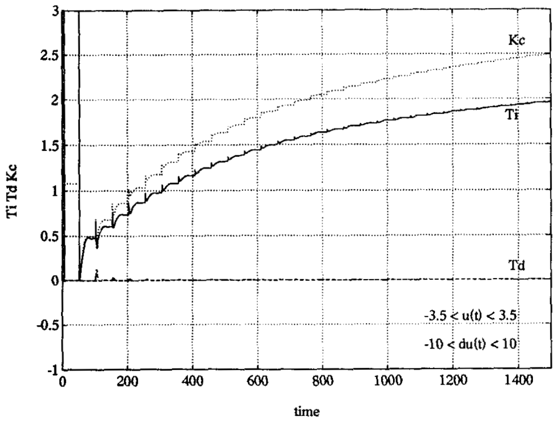

Figure 4: Autotuning with control signal saturation, PID parameters.

Increasing the signal such that the control signal requirement exceeds the physical limitations, the algorithm saturates the control signal over a considerable period. The autotuner interprets the saturation as a lower system gain. Thus Ti converges to a lower value, corresponding to higher controller gain at low frequencies. Convergence is considerably slower, as can be expected.

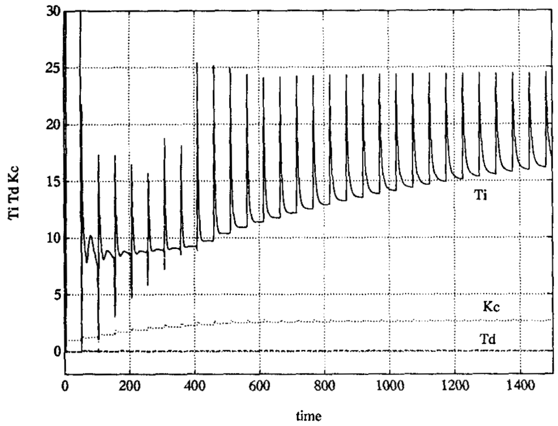

Figure 5: Control signal limits included as constraints, PID parameters.

The spikes in Ti are projections caused by the limitation of the control signal. Experiments over a considerably longer period indicates convergence, which is now very slow. The gain of the tuned controller is lower than in fig 2, but is closer than in fig 4. The spike problem is caused by the high frequency contents of the square reference signal. If this reference signal is known, the problem can be overcome by limiting the bandwidth of the predictors using a reference model. This model can be chosen as a first order model, that with the given square input does not saturate the control signal. In this case, the autotuner does not decrease the controller gain, and the parameters now resembles the ones of figure 2. Convergence speed is determined by the frequency content of the predictor outputs and thus somewhat slower than in the unconstrained case.

The values obtained from the illustrated experiments at time T_{max} are given in table 1.

Experiment	T_{max}	K_c	T_i	T_d
Coupled tanks. without constraints	1500	4.17	14.10	0
Coupled tanks. control signal saturation	1500	2.5	1.96	0
Coupled tanks. Constrained control signal included	1500	2.67	16.95	0.05

Table 1. Simulation results. Coupled tanks with N = 15 and $\alpha = 0.1$.

Conclusions

We have introduced a method for automatic tuning of a ideal PID controller. The optimal controller parameters are obtained through minimization of the GPC criterion. The robustness properties of the GPC are inherited by the design. Physical signal constraints are incorporated in the minimization. The algorithm offers a wide range of tuning parameters, increasing the class of controllable systems, but making the controller more difficult to tune. We stress the fact, that the implementor is free to choose in this well-known applicability/complexity tradeoff.

Facing a timevarying process the proposed method is easily modified for use as an adaptive PID controller.

Simulation studies for a system of coupled tanks have indicated that the method performs well, and that signal limitations can be included in a straightforward manner.

References

[1] Hägglund,T. and K.J. Åström, An Industrial Adaptive PID Controller, Proc. Adaptive systems in control and signal processing, Glasgow 1989.

[2] Cameron,F. and D.E. Seborg, A self-tuning controller with a PID structure, Int. J. Control, 38, pp.401, 1983.

[3] Vega,P. and M.B. Zarrop, Optimal approaches to self-tuning PID control, Proc. Adaptive systems in control and signal processing, Glasgow 1989.

[4] Clarke,D.W. , C. Mohtadi and P.S. Tuffs, Generalized Predictive Control, Automatica, Vol 23, No. 2 pp 137, 1987.

[5] Goodwin,G.C. and K.S. Sin, Adaptive Filtering, Prediction and Control, Prentice Hall, Englewood Cliffs, NJ, 1984.

[6] Fernandes,J.M. , C.E. De Souza and G.C. Goodwin, Adaptive control of a coupled tank apparatus, Int. J. Adaptive Control and Signal Processing, Vol. 3, 319, 1989.

[7] Fletcher,R, A general quadratic programming algorithm, J. IMA, vol. 7, pp 76, 1971.

[8] Madsen,K. , Optimering under bibetingelser (in danish), 3.rd edition, 1983.